TITLE **PERFORMANCE OF CONJUGATE GRADIENT-LIKE ALGORITHMS IN TRANSIENT TWO-PHASE SUBCHANNEL ANALYSIS**

AUTHOR(S) **John A. Turner**
J. Michael Doster

Los Alamos

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

PERFORMANCE OF CONJUGATE GRADIENT-LIKE ALGORITHMS IN TRANSIENT TWO-PHASE SUBCHANNEL ANALYSIS

JOHN A. TURNER* AND J. MICHAEL DOSTER[†]

Abstract. A transient, drift-flux subchannel analysis code (SWIRL) has been created for the development and evaluation of algorithms for the solution of weakly three-dimensional fluid flow problems. Spatial discretization on a staggered grid, semi-implicit temporal discretization, and algebraic reduction of the conservation equations of mass, energy, and momentum result in nonsymmetric block-tridiagonal linear systems of equations that must be solved for the pressure distribution at each time step of a transient. The solution of these systems of equations is the most time-consuming portion of the code, and direct, stationary iterative, and preconditioned conjugate gradient (CG)-like methods have been investigated both for a simple approach to steady-state and for a severe transient. The best direct algorithm appears to be an efficient implementation of block elimination, and iterative methods are compared to this algorithm for accuracy, robustness, and efficiency. Results presented here indicate that preconditioned CG-like methods such as Sonneveld's conjugate gradients squared are superior to an efficient direct method.

Key Words. Subchannel analysis, drift-flux, semi-implicit, nonsymmetric linear systems, block-tridiagonal linear systems, preconditioned conjugate gradients, conjugate gradients squared.

1. Introduction. Subchannel analysis is arguably the most widely used method of modelling the thermal-hydraulic behaviour of fluid flow parallel to rod bundles, as occurs in the cores of nuclear reactors. In this approach, the flow field is divided into a number of quasi-one-dimensional subchannels that communicate laterally by crossflow due to both non-uniformities in the radial pressure distribution and turbulent mixing. Examples of codes based on the subchannel analysis philosophy abound, the most well-known being the many versions of the Battelle Pacific Northwest code COBRA ([26, 27, 28, 29, 2, 42, 15]), the COBRA extension VIPRE [39, 40], SABRE-1 [25], the French code FLICA [13], the British code HAMBO [3], and the Westinghouse code THINC [43, 4]. A major strength of the method is its versatility. Subchannels are defined only by flow area and wetted perimeter, so the exact shape of the channel is not needed, and lateral connections are defined by width and distance between subchannel centroids. Since there is no momentum coupling of crossflows, discrete lateral coordinates and lateral boundary conditions are not needed. A crossflow exists and has its direction defined by the two subchannels it connects. As a result, triangular and hexagonal lattices can be simulated as easily as square lattices.

It should be mentioned that other approaches have been used for rod-bundle thermal-hydraulic analysis. Sha [33] compares subchannel analysis to the porous medium formulation used in the COMMIX series of codes [34, 35, 9, 8] and the boundary-fitted coordinate system approach used in the BODYFIT codes [36, 6, 5]. Most complex is the full three-dimensional, six-equation, two-fluid approach used in THERMIT [22], which solves the same equations as TRAC P1A [23].

These codes are, for the most part, large, mature production codes that have been developed over many years. It is thus often difficult to to significantly restructure these codes in order to investigate alternative solution algorithms or performance on advanced architecture computers. For this reason, a new code, SWIRL [41], has been developed to aid in this type of investigation.

2. SWIRL Numerics. SWIRL uses a three-equation drift-flux model, in which thermal equilibrium is assumed and slip is treated as a diffusion-like phenomenon [44]. The conservation equations of mass, energy, and momentum are reduced to a system of linear algebraic equations in pressure as follows:

* Los Alamos National Laboratory, Reactor Design and Analysis Group (N 12), MS K551, Los Alamos, NM, 87545, jat@jatstrat ta52 lanl gov

† North Carolina State University, Department of Nuclear Engineering, Box 7909, Raleigh, NC, 27695

1. Discretize spatially and integrate over appropriate control volumes to obtain a set of ordinary differential equations.
2. Reduce to a set of linear algebraic equations by using a semi-implicit time advancement scheme.
3. Algebraically reduce to a single system of linear equations in pressure only.

2.1. Spatial Discretization. The conservation equations of mass, internal energy, and momentum are:

Continuity

$$\frac{\partial \rho}{\partial t} + \overrightarrow{\nabla} \cdot (\rho \vec{v}) = 0 \tag{1}$$

Internal Energy

$$\frac{\partial}{\partial t}(\rho u) + \overrightarrow{\nabla} \cdot (\rho u \vec{v}) = -P\overrightarrow{\nabla} \cdot \vec{v} + \overrightarrow{\nabla} \cdot \vec{q} \tag{2}$$

Axial Momentum

$$\frac{\partial}{\partial t}(\rho V) + \overrightarrow{\nabla} \cdot (\rho V \vec{v}) = -\frac{\partial P}{\partial z} + \overrightarrow{\nabla} \cdot \vec{\sigma_z} + \rho g \tag{3}$$

Lateral Momentum

$$\frac{\partial}{\partial t}(\rho w) + \overrightarrow{\nabla} \cdot (\rho w \vec{v}) = -\frac{\partial P}{\partial r} + \overrightarrow{\nabla} \cdot \vec{\sigma_r} . \tag{4}$$

Now consider the computational grid in **Figure 1** showing three axial levels of two adjacent subchannels. Following standard practices (see [21] and [10], for example), a staggered mesh is defined such that scalar quantities (density, pressure, void, *etc.*) are defined at the centers of computational cells ($i, k-1$; $j, k-1$; ik; jk; $i, k+1$; and $j, k+1$ in Figure 1) and vector quantities (velocity, mass flux, *etc.*) are defined at cell boundaries ($i, k-\frac{1}{2}$; $j, k-\frac{1}{2}$; $i, k+\frac{1}{2}$; and $j, k+\frac{1}{2}$ for axial mass flux, and $ij, k-1$; ijk; and $ij, k+1$ for lateral mass flux).

The continuity and internal energy equations are integrated over, for example, computational cell ik, bounded axially by $i, k-\frac{1}{2}$ and $i, k+\frac{1}{2}$, and the axial momentum equation over, for example, computational cell $i, k+\frac{1}{2}$, bounded axially by ik and $i, k+1$, define average quantities, and approximate the products of averages with the averages of products. The control volume for lateral momentum is centered axially and staggered laterally, so the lateral momentum equation is integrated over, for example, cell ijk, with axial boundaries $ij, k-\frac{1}{2}$ and $ij, k+\frac{1}{2}$ and lateral boundaries at ik and jk.

Integrating the four equations over the appropriate control volumes, defining appropriate mixture quantities, and rearranging to include drift flux terms yields the following set of first-order, non-linear, ordinary differential equations:

Continuity

$$\Delta z \frac{d\rho_{i,k}}{dt} + \left\{ (\rho V)_{i,k+\frac{1}{2}} - (\rho V)_{i,k-\frac{1}{2}} \right\} + \frac{1}{A_{r_i}} \sum_j (A_c)_{ij}(\rho w)_{ij,k} = 0 . \tag{5}$$

Internal Energy

$$\Delta z \frac{d}{dt}(\rho u)_{ik} + \left\{ (\rho u V)_{i,k+\frac{1}{2}} - (\rho u V)_{i,k-\frac{1}{2}} \right\} +$$

2

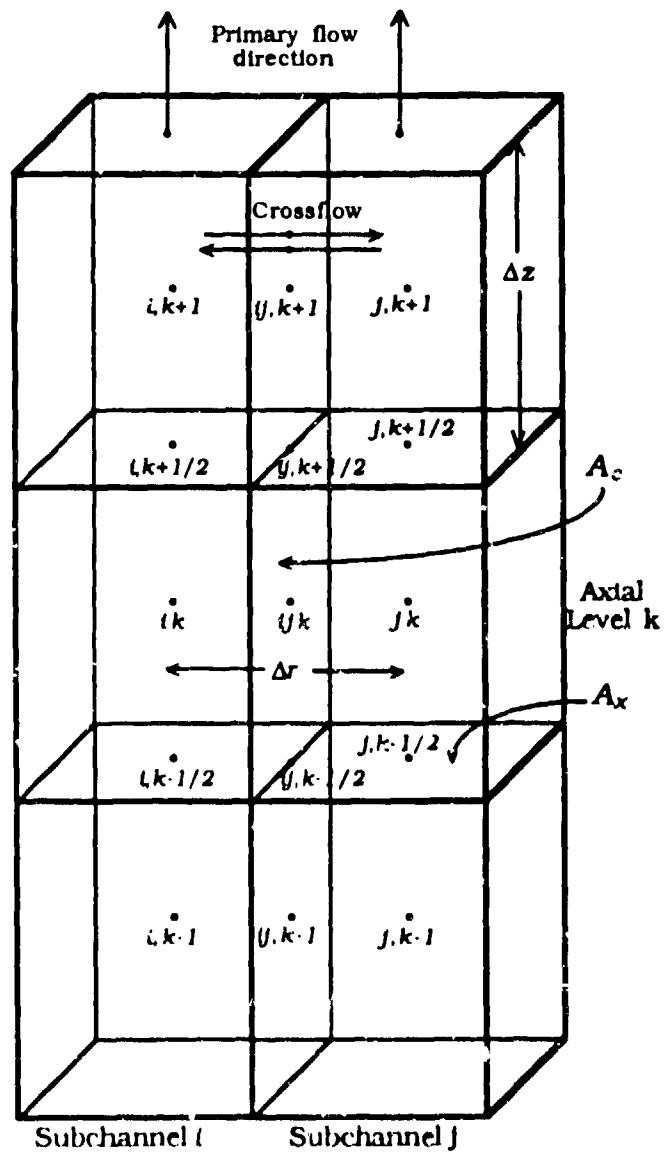FIG. 1. *Section of computational grid.*

$$+\left[\frac{\tilde{\rho}_l\tilde{\rho}_g}{\tilde{\rho}}(\tilde{u}_g-\tilde{u}_l)j_D^g\right]_{i,k+\frac{1}{2}}-\left[\frac{\tilde{\rho}_l\tilde{\rho}_g}{\tilde{\rho}}(\tilde{u}_g-\tilde{u}_l)j_D^l\right]_{i,k-\frac{1}{2}}+$$

$$+\frac{1}{A_{x_i}}\sum_j(A_c)_{ij}(\rho w)_{ijk}u_{ijk}^*+\frac{1}{A_{x_i}}\sum_j(A_c)_{ij}\left[(\rho u)_{ik}-(\rho u)_{jk}\right]w'_{ijk}=$$

$$-P_{ik}\left[j_{i,k+\frac{1}{2}}-j_{i,k-\frac{1}{2}}+\frac{1}{A_{x_i}}\sum_j(A_c)_{ij}w_{ijk}\right]+\frac{Q_{ik}}{A_{x_i}},\tag{6}$$

where

$$u_{ijk}^*=\begin{cases}u_{ik}&\text{if }(\rho w)_{ijk}>0\\u_{jk}&\text{if }(\rho w)_{ijk}<0\end{cases}.\tag{7}$$

and w' is the crossflow velocity due to turbulent mixing. The assumption is that crossflow is separable into flow due to pressure imbalances between subchannels and flow due to turbulent mixing. This is not strictly a turbulence model, but simply an empirical attempt to account for turbulent exchange of energy and momentum. Turbulence is assumed to result in no net mass exchange.

Axial Momentum

$$\Delta z\frac{d}{dt}(\rho V)_{i,k+\frac{1}{2}}+\left\{(\rho V^2)_{i,k+1}-(\rho V^2)_{ik}\right\}+$$

$$+\left[\frac{\rho_l\rho_g}{\rho}\left(\frac{1}{\alpha_l\alpha_g}\right)\left(\tilde{j}_D^g\right)^2\right]_{i,k+1}-\left[\frac{\rho_l\rho_g}{\rho}\left(\frac{1}{\alpha_l\alpha_g}\right)\left(\tilde{j}_D^g\right)^2\right]_{ik}+$$

$$+\frac{1}{A_{x_i}}\sum_j(A_c)_{ij}(\rho Vw)_{ij,k+\frac{1}{2}}+\frac{1}{A_{x_i}}\sum_j(A_c)_{ij}\left[(\rho V)_{i,k+\frac{1}{2}}-(\rho V)_{j,k+\frac{1}{2}}\right]w'_{ij,k+\frac{1}{2}}=$$

$$-(P_{i,k+1}-P_{ik})-\left[\frac{(\tau_w P_w)_{i,k+\frac{1}{2}}}{A_{x_i}}\right]_{2\phi}\Delta z+\tilde{\rho}_{i,k+\frac{1}{2}}g\Delta z,\tag{8}$$

where

$$(\rho V^2)_{ik}=\frac{(\rho\bar{V})_{ik}(\rho\tilde{V})_{ik}}{\rho_{ik}},\tag{9}$$

$$(\rho\bar{V})_{ik}=\frac{1}{2}\left[(\rho V)_{i,k-\frac{1}{2}}+(\rho V)_{i,k+\frac{1}{2}}\right],\tag{10}$$

and

$$(\rho\tilde{V})_{ik}\quad\begin{cases}(\rho V)_{i,k-\frac{1}{2}}&\text{if }(\rho V)_{ik}>0\\(\rho V)_{i,k+\frac{1}{2}}&\text{if }(\rho V)_{ik}<0\end{cases}.\tag{11}$$

4

Similarly,

$$(\rho V w)_{ij,k+\frac{1}{2}} = (\tilde{\rho w})_{ij,k+\frac{1}{2}} V_{ij,k+\frac{1}{2}} \ , \tag{12}$$

where

$$V_{ij,k+\frac{1}{2}} = \frac{(\bar{\rho V})_{ij,k+\frac{1}{2}}}{\tilde{\rho}_{ij,k+\frac{1}{2}}} \ , \tag{13}$$

$$(\bar{\rho V})_{ij,k+\frac{1}{2}} = \frac{1}{2}\left[(\rho V)_{i,k+\frac{1}{2}} + (\rho V)_{j,k+\frac{1}{2}}\right] \ , \tag{14}$$

$$\tilde{\rho}_{ij,k+\frac{1}{2}} = \frac{1}{2}\left(\tilde{\rho}_{i,k+\frac{1}{2}} + \tilde{\rho}_{j,k+\frac{1}{2}}\right) \ , \tag{15}$$

and

$$(\tilde{\rho w})_{ij,k+\frac{1}{2}} = \begin{cases} (\rho w)_{ijk} & \text{if } V_{ij,k+\frac{1}{2}} > 0 \\ (\rho w)_{ij,k+1} & \text{if } V_{ij,k+\frac{1}{2}} < 0 \end{cases} \ . \tag{16}$$

For the frictional pressure drop term, the standard procedure of multiplying the equivalent saturated single-phase loss by a semi-empirically determined two-phase multiplier, ϕ_{lo}^2, is used. Thus,

$$\left[\frac{(\tau_w P_w)_{i,k+\frac{1}{2}}}{A_{x_i}}\right]_{2\phi} = -\left[\phi_{lo}^2 \frac{(\rho V)^2}{2\tilde{\rho}_l}\left(\frac{f_{1\phi}}{D_H} + \frac{k}{\Delta z}\right)\right]_{i,k+\frac{1}{2}} \ . \tag{17}$$

Lateral Momentum

$$\Delta z \frac{d}{dt}(\rho w)_{ij,k} + \left\{(\rho V w)_{ij,k+\frac{1}{2}} - (\rho V w)_{ij,k-\frac{1}{2}}\right\} =$$

$$-\left(\frac{A_c}{A_{r_s}}\right)_{ij}(P_{jk} - P_{ik}) - \left[\frac{(\tau_w P_w)_{ijk}}{(A_{r_s})_{ij}}\right]_{2\phi}\Delta r_{ij} \ . \tag{18}$$

Following VIPRE methodology, pressure loss through the gap is computed using an overall loss coefficient k_r due to the complex geometry of the gap and the difficulty in determining an appropriate lateral friction factor [39]. The coefficient accounts for friction as well as forms loss due to the area change. The flow is assumed to be homogeneous in the lateral direction, and, as in the axial momentum equation a, two-phase friction multiplier is used to compute the lateral pressure drop. Thus,

$$\left[\frac{(\tau_w P_w)_{ijk}}{(A_{r_s})_{ij}}\right]_{2\phi} = -\left[\phi_{lo}^2 \frac{(\rho w)^2}{2(\rho_l)^*}\frac{k_r}{\Delta r}\right]_{ijk} \ , \tag{19}$$

where $(\rho_l)_{ijk}^*$ is laterally donored as in Equation 7 for u_{ijk}^*.

Note the following assumptions inherent in the above equations (in addition to those previously mentioned):

1. Axial flow areas are constant along a subchannel.
2. The Zuber-Findley void quality relation holds in the axial direction.
3. Heat conduction through the fluid is negligible.
4. Axial turbulent mixing is negligible.
5. Crossflow is small relative to axial flow, so that the $(\rho w w')$ term in the lateral momentum equation can be neglected.

2.2. Temporal Discretization.

Temporal discretization is semi–implicit; *i.e.*, time derivatives are expanded such that the resulting equation set is linear in the new–time variables (see [21] and [10], for example). With some exceptions, pressures and velocities (or in this case mass fluxes) are evaluated at the new time level and convected terms at the past time level. This scheme imposes a time–step limit based on the material transport time, or Courant limit [21], and results in the following system of linear algebraic equations, in which all terms without a time level superscript are evaluated at the present time level.

Continuity

$$\frac{\Delta z}{\Delta t}(\rho_{ik}^{t+\Delta t} - \rho_{ik}) + \left\{(\rho V)_{i,k+\frac{1}{2}}^{t+\Delta t} - (\rho V)_{i,k-\frac{1}{2}}^{t+\Delta t}\right\} + \frac{1}{A_{x_i}}\sum_j (A_c)_{ij}(\rho w)_{ijk}^{t+\Delta t} = 0 . \tag{20}$$

Internal Energy

$$\frac{\Delta z}{\Delta t}\left[(\rho u)_{ik}^{t+\Delta t} - (\rho u)_{ik}\right] + \left\{(\rho V)_{i,k+\frac{1}{2}}^{t+\Delta t}\tilde{u}_{i,k+\frac{1}{2}} - (\rho V)_{i,k-\frac{1}{2}}^{t+\Delta t}\tilde{u}_{i,k-\frac{1}{2}}\right\} +$$

$$+ \left[\frac{\tilde{\rho}_l\tilde{\rho}_g}{\tilde{\rho}}(\tilde{u}_g - \tilde{u}_l)(j_D^g)\right]_{i,k+\frac{1}{2}} - \left[\frac{\tilde{\rho}_l\tilde{\rho}_g}{\tilde{\rho}}(\tilde{u}_g - \tilde{u}_l)(j_D^g)\right]_{i,k-\frac{1}{2}} +$$

$$+ \frac{1}{A_{x_i}}\sum_j (A_c)_{ij}(\rho w')_{ijk}^{t+\Delta t}u_{ijk}^* + \frac{1}{A_{x_i}}\sum_j (A_c)_{ij}[(\rho u)_{ik} - (\rho u)_{jk}]w'_{ijk} =$$

$$- P_{ik}\left[j_{i,k+\frac{1}{2}} - j_{i,k-\frac{1}{2}} + \frac{1}{A_{x_i}}\sum_j (A_c)_{ij}w_{ijk}\right] + \frac{q_{ik}^{t+\Delta t}}{A_{x_i}} . \tag{21}$$

Axial Momentum

$$\frac{\Delta z}{\Delta t}\left[(\rho V)_{i,k+\frac{1}{2}}^{t+\Delta t} - (\rho V)_{i,k+\frac{1}{2}}\right] + \left\{(\rho V^2)_{i,k+1} - (\rho V^2)_{ik}\right\} +$$

$$+ \left[\frac{\mu_l\rho_g}{\rho}\left(\frac{1}{\alpha_l\alpha_g}\right)\left(\tilde{j}_D^g\right)^2\right]_{i,k+1} - \left[\frac{\rho_l\rho_g}{\rho}\left(\frac{1}{\alpha_l\alpha_g}\right)\left(\tilde{j}_D^g\right)^2\right]_{ik} +$$

$$+ \frac{1}{A_{x_i}}\sum_j (A_c)_{ij}(\rho V w')_{ij,k+\frac{1}{2}} + \frac{1}{A_{x_i}}\sum_j (A_c)_{ij}\left[(\rho V)_{i,k+\frac{1}{2}} - (\rho V)_{j,k+\frac{1}{2}}\right]w'_{ij,k+\frac{1}{2}} =$$

$$- (P_{i,k+1}^{t+\Delta t} - P_{ik}^{t+\Delta t}) - \left[\frac{(\tau_w P_w)_{i,k+\frac{1}{2}}}{A_{x_i}}\right]_{2\phi}^{t^*}\Delta z + \tilde{\rho}_{i,k+\frac{1}{2}}g\Delta z \tag{22}$$

and

$$\left[\frac{(\tau_w P_w)_{i,k+\frac{1}{2}}}{A_{x_i}}\right]_{2\phi}^{t^*} = \left[\phi_{lo}^2\frac{|(\rho V)|}{2\tilde{\rho}_i}\left(\frac{f_{l\phi}}{D_H} + \frac{k}{\Delta z}\right)\right]_{i,k+\frac{1}{2}}(\rho V)_{i,k+\frac{1}{2}}^{t+\Delta t} . \tag{23}$$

6

Lateral Momentum

$$\frac{\Delta z}{\Delta t}\left[(\rho w)_{ijk}^{t+\Delta t} - (\rho w)_{ijk}\right] + \left\{(\rho V w)_{ij,k+\frac{1}{2}} - (\rho V w)_{ij,k-\frac{1}{2}}\right\} =$$

$$-\left(\frac{A_c}{A_{r_s}}\right)_{ij}(P_{jk}^{t+\Delta t} - P_{ik}^{t+\Delta t}) - \left[\frac{(\tau_w P_w)_{ijk}}{(A_{r_x})_{ij}}\right]_{2\phi}^{t^*}\Delta r_{ij} \tag{24}$$

where, similar to the al⋯,

$$\left[\frac{(\tau_w P_w)_{ijk}}{(A_{r_s})_{ij}}\right]_{2\phi}^{t^*} = \left[\phi_{lo}^2 \frac{|(\rho w)|}{2\tilde{\rho}_l}\frac{k_r}{\Delta r}\right]_{ijk}^{t^*}(\rho w)_{ijk}^{t+\Delta t} . \tag{25}$$

2.3. Equation of State. As in RELAP5/MOD1 [12], an equation of state of the form $\rho = \rho(\rho u, P)$ provides closure. To preserve the linearity of the rest of the equation set, the density relationship is linearized by expanding in a two–term Taylor series expansion about the old time level. Following RELAP5/MOD1 methodology,

$$\rho^{t+\Delta t} = \rho^t + \left[(\rho u)^{t+\Delta t} - (\rho u)^t\right]\left.\frac{\partial\rho}{\partial\rho u}\right|_{u,P} + (P^{t+\Delta t} - P^t)\left.\frac{\partial\rho}{\partial P}\right|_{u,\rho u} . \tag{26}$$

2.4. Algebraic Reduction. The system of linear algebraic equations given in Equations 20 through 24 could be posed as a matrix equation and solved for all new time variables (P, ρ, ρu, ρV, and ρw) at once. However, the size of the system can be reduced and the structure of the coefficient matrix improved if four of the new time variables are eliminated in favor of the remaining one. This approach adds a back substitution step following the solution of the linear system to obtain the remaining new time level values.

This results in a single system in the spatial pressure distribution at the new time level:

$$a_{i,k-\frac{1}{2}}P_{i,k-1}^{t+\Delta t} + b_{ik}P_{ik}^{t+\Delta t} + a_{i,k+\frac{1}{2}}P_{i,k+1}^{t+\Delta t} + \sum_{j}c_{ijk}P_{jk}^{t+\Delta t} = S_{ik} . \tag{27}$$

The first three terms of Equation 27 represent the pressure distribution in the axial direction and the summation term represents communication in the lateral direction. The coefficient matrix can be shown to be block–tridiagonal, with diagonal off–diagonal blocks. For example, consider the 1/8th section of an 8 rod by 8-rod BWR assembly shown in Figure 2. For a subchannel analysis model with three axial nodes the coefficient matrix has the sparsity pattern shown in Figure 3 (shaded squares represent non–zero elements). Note that the bandwidth of the main–diagonal blocks is equal to the maximum difference in the indices of any two adjacent subchannels (in this case four). For a rectangular array the main–diagonal blocks become five–banded, and in the general case the main diagonal blocks are diagonal with unstructured off–diagonal elements. Note that the matrix is geometrically but not numerically symmetric, and that each main diagonal block represents a two dimensional axial slice across the channel. If N_{chan} denotes the number of subchannels and N_{ax} the number of axial nodes, the size of the system is

$$(N_{chan})(N_{ax}) \text{ by } (N_{chan})(N_{ax}) . \tag{28}$$

A typical BWR fuel assembly has 64 fuel rods in an 8 by 8 array, yielding 81 subchannels. Use of 48 axial cells (which results in approximately 3 inch cells) requires solution of a linear system with 3888 unknowns at each time step (forty eight 81 by 81 blocks).
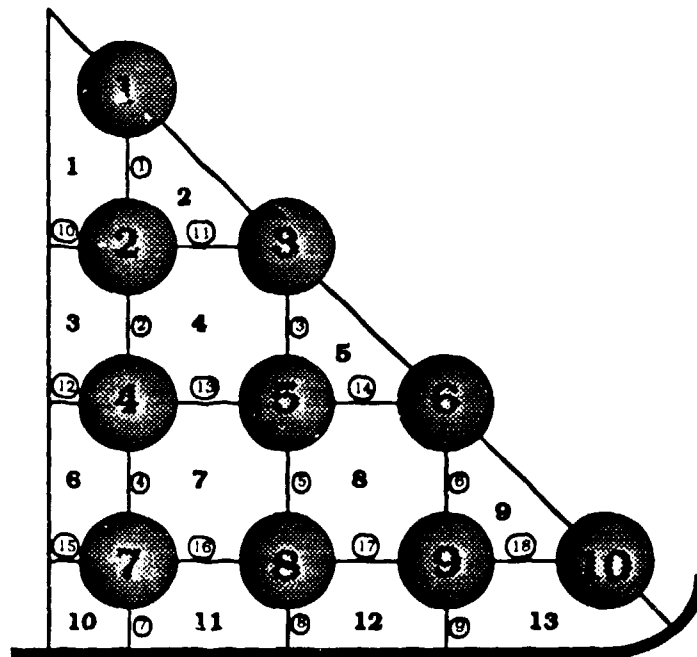
7

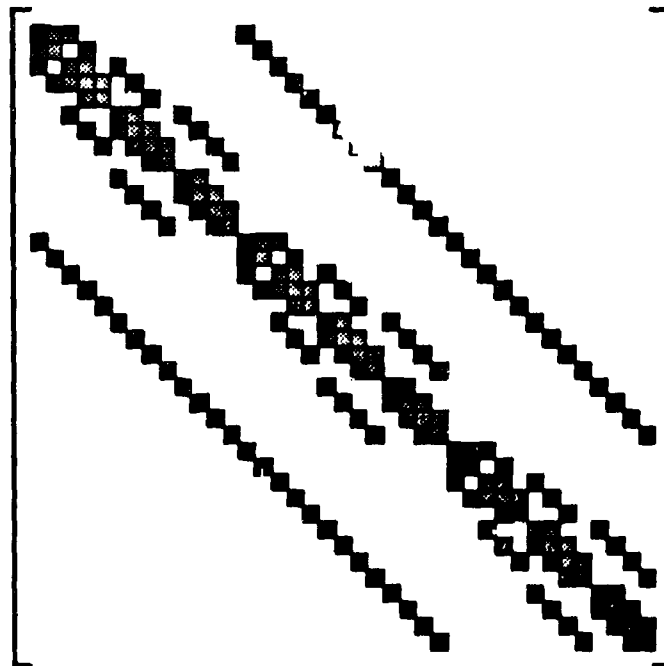FIG. 2. *Subchannel analysis model of a 1/8th section of an 8-rod by 8-rod BWR assembly.*

FIG. 3 *Structure of coefficient matrix for a subchannel analysis model of a 1/8th section of an 8-rod by 8-rod BWR assembly using three axial nodes*

$$\begin{bmatrix} A_1 & C_1 \\ B_1 & A_2 & C_2 \\ & B_2 & A_3 & \ddots \\ & & \ddots & \ddots & C_{N_{ax}-1} \\ & & & B_{N_{ax}-1} & A_{N_{ax}} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ \vdots \\ X_{N_{ax}} \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ \vdots \\ S_{N_{ax}} \end{bmatrix}$$

FIG. 4. *Block form of matrix system.*

$D_1 = A_1^{-1}C_1$, $E_1 = A_1^{-1}S_1$

For $n = 2$ to $N_{ax} - 1$:

$\quad D_n = (A_n - B_{n-1}D_{n-1})^{-1}C_n$

$\quad E_n = (A_n - B_{n-1}D_{n-1})^{-1}(S_n - B_{n-1}E_{n-1})$

$E_{N_{ax}} = (A_{N_{ax}} - B_{N_{ax}-1}D_{N_{ax}-1})^{-1}(S_{N_{ax}} - B_{N_{ax}-1}E_{N_{ax}-1})$

$X_{N_{ax}} = E_{N_{ax}}$

For $n = N_{ax} - 1$ to 1:

$\quad X_n = E_n - D_n X_{n+1}$

FIG. 5. *Block elimination algorithm.*

2.5. Time–step Control. Time–step control is implemented to maintain stability of the solution and to control linearization error in the solution (the error introduced by linearizing the state equation). To control the error in the solution the difference between the mixture density as computed by the system and the density as computed by the equation of state is monitored. Stability is assured if both the lateral and axial Courant limits are satisfied for the duration of the simulation. The lateral Courant limit often becomes the most restrictive for simulations with even moderate crossflow since lateral computational cells are often small. However, we have found that the lateral Courant limit can be relaxed completely, and the system will converge to identical steady–state conditions even for severe flow blockages. This improves run times significantly.

3. Solving Nonsymmetric Block–Tridiagonal Linear Systems.

3.1. Direct Methods. The most efficient direct solver to date is a variant of block elimination, which can be considered an extension of the classic algorithm for tridiagonal systems. If the block system is indexed as shown in Figure 4, such an extension is shown in Figure 5, in which A_n, B_n, and C_n are the nth main–diagonal, sub–diagonal, and super–diagonal blocks of the coefficient matrix, respectively, and X_n and S_n are the nth blocks of the solution and source vectors, respectively. One implementation used in SWIRL, referred to as BTD_{inv} , takes advantage of the diagonal off–diagonal blocks in the matrix multiplications and performs the inversions explicitly. The resulting operation count for BTD_{inv} is $\mathcal{O}(2N_{ax}N_{chan}^3)$.

BTD_{inv} has proven to be superior to other direct methods both on single processor and few–CPU, shared memory, pipelined vector machines. For a discussion of other implementations of block elimination and other direct methods such as banded elimination, cyclic reduction, and parallel partition methods see [41].

3.2. Conjugate Gradient–Like Methods. Conjugate gradient (CG) and CG–like methods have seen renewed interest in recent years, particularly when combined with appropriate preconditioning schemes. Though the classic CG algorithm is only applicable to the solution of linear systems with symmetric positive definite coefficient matrices, many CG like approaches have been

9

$$w = b - Ax_0$$

solve $Mr_0 = w$

$$q = u = p_0 = r_0$$

$$\rho_0 = (q, r_0)$$

for $i = 0, 1, \ldots,$ until convergence:

  solve $Mw = Ap_i$

  $\alpha = \rho_i/(q, w)$

  $h = u - \alpha w$

  $u = (u + h)\alpha$

  $x_{i+1} = x_i + u$

  solve $Mw = Au$

  $r_{i+1} = r_i - w$

  $err = \|r_{i+1}\|_\infty / \|x_{i+1}\|_\infty$

  if $err < \epsilon \implies$ converged

  $\rho_{i+1} = (q, r_{i+1})$

  $\beta = \rho_{i+1}/\rho_i$

  $u = r_{i+1} + \beta h$

  $p_{i+1} = u + \beta(\beta p_i + h)$

FIG. 6. *Conjugate gradients squared (CGS) algorithm (combined and modified form of versions given in [38], [37], and [7]).*

developed for nonsymmetric linear systems. Saad has classified many of these methods into the following four subgroups within the larger group of Kryiov subspace methods [31]:

1. The full orthogonalization method (FOM) [30], ORTHORES, and Axelsson's method [1]. When the coefficient matrix is symmetric, standard CG falls into this catagory.
2. The conjugate residual method (e.g. [17] and [18]) and GMRES [32].
3. Bi-conjugate gradients (BCG) [14] and conjugate gradients squared (CGS) [37].
4. CG applied to the normal equations $(A^T Ax = A^T b)$.

For this work. Sonneveld's CGS algorithm, shown in Figure 6, is used. Note that $Ax = b$ is the system being solved, x_0 is an initial guess for the solution vector (taken to be the pressure distribution at the past time step), M is the preconditioning matrix (a matrix which is "easy" to factor and in some way "close" to A), and r, are residual vectors. The ratio of the infinity norms of the residual and solution vector is used as a stopping criterion, as in [7]. A complete description of the algorithm and its relationship to other CG methods is given in [37]. CGS has been shown to perform well in comparison with other CG-like methods in several recent papers [37, 20, 19, 16].

CGS has been implemented in SWIRL for the special case of rectangular lattices, which, as explained in Section 2.4, result in seven-banded coefficient matrices. This allows the use of efficient matrix-vector multiplications in the algorithm. This is not as restrictive as it might seem at first, since even in the general case there are never more than seven nonzero elements in a row or column of the coefficient matrix (as long as each subchannel is connected to at most four other subchannels), so seven-banded matrix vector multiplication could be used even in that case, but indexing would be somewhat more complicated. Using simple diagonal preconditioning with this implementation yields the algorithm $CGS7_d$, with an operation count of $\mathcal{O}(42 N_{ax} N_{chan} N_{iter})$. This implies that the number of iterations required to beat BTD_{inv} is $\mathcal{O}(N_{chan}^2/21)$.

Results will be presented for a more innovative preconditioning scheme, however. Early in this work it was observed that even though the elements of the coefficient matrix change at each step in a transient and thus must be recomputed at each step, the spectral radii of iteration matrices

$$A_1^* = A_1^{-1}$$
$$D_1 = A_1^* C_1$$
$$\text{For } n = 2 \text{ to } N_{ax} - 1:$$
$$A_n^* = (A_n - B_{n-1} D_{n-1})^{-1}$$
$$D_n = A_n^* C_n$$

FIG. 7. *Factorization stage of block elimination algorithm.*

$$E_1 = A_1^* S_1$$
$$\text{For } n = 2 \text{ to } N_{ax}:$$
$$E_n = A_n^* (S_n - B_{n-1} E_{n-1})$$
$$X_{N_{ax}} = E_{N_{ax}}$$
$$\text{For } n = N_{ax} - 1 \text{ to } 1:$$
$$X_n = E_n - D_n X_{n+1}$$

FIG. 8. *Elimination stage of block elimination algorithm.*

for stationary iterative methods remain essentially constant throughout a transient. Though the convergence of CG-like methods is dependent on properties ̇e eigenspectrum other than the spectral radius, it seems appropriate to attempt to make use ̇us property.

One way to accomplish this is to factor the coefficient matrix at one time step, use that factorization to solve for the pressure at that time step, the use the factorization (or part of it) as the preconditioner for CGS in future advancement attempts. The danger in this approach is that in a severe transient the coefficient matrix might change enough to cause CGS to be very inefficient (or to fail to converge). In an attempt to minimize such effects this strategy was implemented so that whenever CGS requires more than a specified number of iterations (which will be denoted L_{iter}), a new preconditioner is obtained at the next advancement. Thus, if this iteration limit were set at zero, SWIRL would alternate between using the direct method and CGS.

To implement this scheme, which will be referred to as *past factorization* preconditioning, recall the block elimination algorithm from Figure 5. Note that the algorithm can be split into a factorization stage and an elimination stage, as shown in Figures 7 and 8. The scheme is then as follows:

1. For the first attempted time step, the linear system is solved by BTD_{inv} and the block factorization, stored in A_n^* and D_n, saved.

2. The coefficient matrix from that attempted advancement is then used as a preconditioner in CGS for subsequent advancement attempts. That is, A_n^* and D_n contain the factorization of the preconditioning matrix M from Figure 6, so solution of the two linear systems in each iteration of CGS ($Mw = Ap$, and $Mw = Au$) require only the forward and backward sweep shown in Figure 8.

3. The number of iterations required for convergence is monitored, and if it exceeds some limit, the next advancement is again solved by BTD_{inv}. In the process the block factorization of a new (and hopefully better) preconditioner is obtained.

We call this PBLU preconditioning, since a Past time step Block LU decomposition is used. When PBLU preconditioning is used with the seven banded version of CGS, the algorithm is called CGS7_b, and has an operation count of $\mathcal{O}((8 N_{ax} N_{chan}^2 + 46 N_{ax} N_{chan}) N_{iter})$, implying a breakeven number of iterations (compared to BTD_{inv}) of $\mathcal{O}(N_{chan}/4)$. Note that a more robust approach might be to monitor the progress of CGS and force use of the direct method if the iteration appears
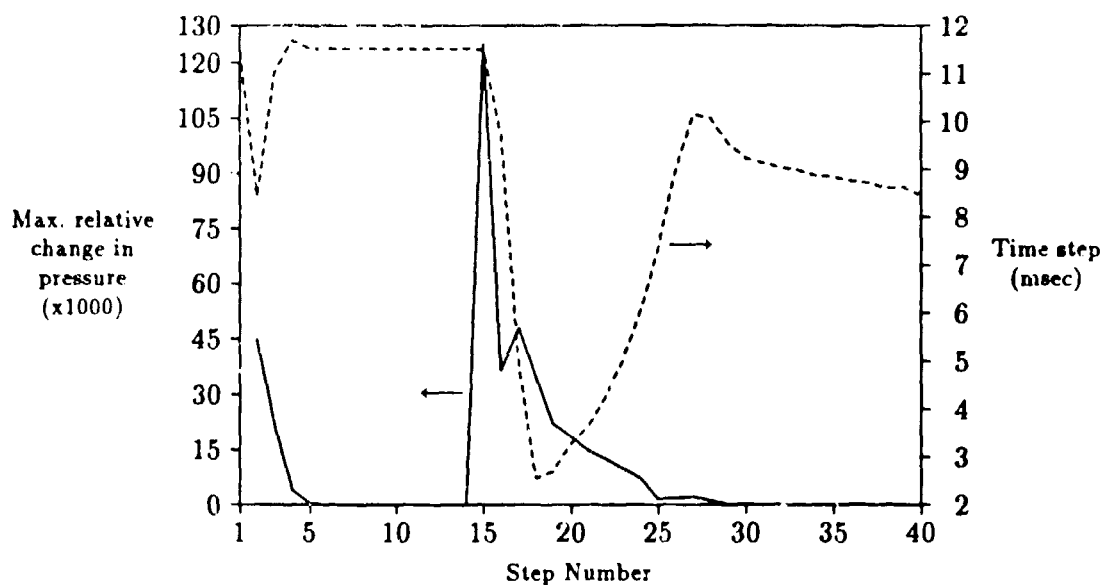
FIG. 9. *Maximum relative change in pressure and time step used at each in the transient described in Section 4.2.*

to be stalled or diverging.

4. Test Problems.

4.1. Approach to Steady–State. For this problem SWIRL is executed in steady–state mode with inlet mass flux / outlet pressure BCs on a PWR-type assembly. That is, the assembly is square and open, with geometrically identical subchannels. Power distribution is uniform radially and a chopped cosine axially, with a peak linear heat additior rate (q') of 5.44 kW/ft, the average value for McGuire Nuclear Station [11]. Note that the fluid remains single phase and that little crossflow occurs under these conditions. The simulation is repeated for assemblies ranging from a 3-rod by 3-rod lattice (4 subchannels and 4 gaps) to a 10 rod by 10-rod lattice (81 subchannels and 144 gaps).

4.2. Severe Transient. The geometry is the same as for the steady–state test case, but only the 10 rod by 10 rod size is used. SWIRL is executed in transient mode with inlet mass flux / outlet pressure BCs and initial conditions set by SWIRL. The maximum normal linear heat addition rate given for McGuire, 12.5 kW/ft, is used [11], and again the power distribution is uniform radially and symmetric axially. At 0.15 seconds into the transient, the following four events occur simultaneously:

1. A flow obstruction of $k = 2$ is introduced in a subchannel near the edge of the assembly, about halfway up the channel ($z = 70$ inches),

2. inlet mass flux undergoes a step decrease of about 20%,

3. outlet pressure drops by almost 15%, and

4. core power uniformly jumps 10%.

The relative change in pressure and the time step used for each advancement are shown in Figure 9, which demonstrates the severity of the transient. The exit void fraction in the channel with the flow obstruction exceeds 0.35.
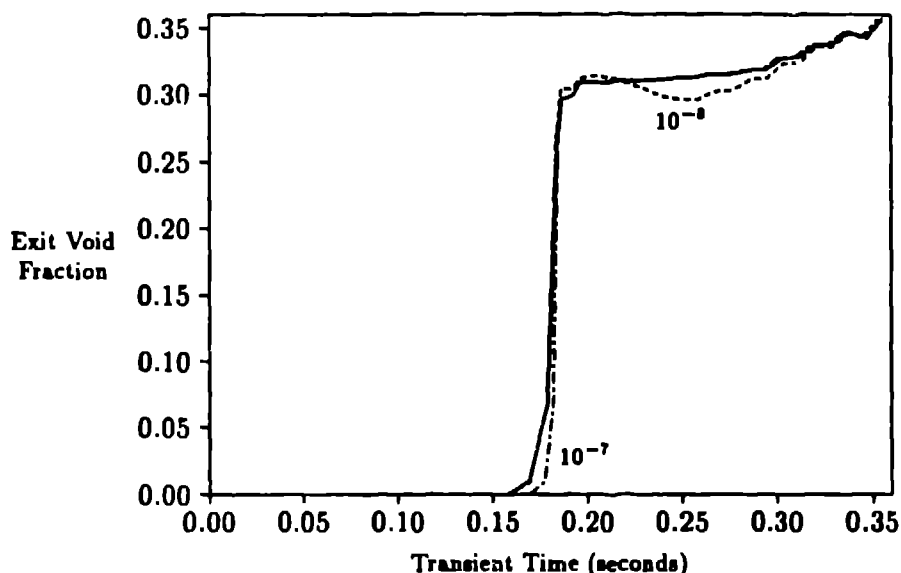
FIG. 10. *Exit void fraction in subchannel 5 during the transient described in Section 4.2 using different convergence criteria in CGS7$_b$ (solid line is for $\epsilon = 10^{-9}$ or tighter and for direct solution).*

5. Results. An Alliant FX/4 with four Computational Elements (CEs) and a 256KB cache was used for the following results. The operating system and compiler versions used were Concentrix 5.0.0 and FX/Fortran 4.2.40, respectively, and compiler options -Ogvc -DAS were used. We have found that although the theoretical peak computational rate for this configuration is nearly 48 MFLOPs, the best that can be expected in a real code (even one that is highly vectorizable and parallelizable by the compiler) is closer to 9 MFLOPs.

5.1. Determination of Parameters for CGS7$_b$. Before CGS7 with PBLU preconditioning can be compared with BTD$_{inv}$, the convergence criterion, ϵ, and the iteration limit, L_{iter}, must be determined. The appropriate convergence criterion was determined first, by using SWIRL / CGS7$_b$ to model the severe transient described in Section 4.2 with convergence criteria of 10^{-7}, 10^{-8}, 10^{-9}, 10^{-10}, and 10^{-11}. Those results were then compared with those obtained from SWIRL/BTD$_{inv}$. The exit void fraction in subchannel 5, shown in Figure 10, indicates that the appropriate value of ϵ is 10^{-9}.

To determine the optimal value of L_{iter}, the convergence criterion was set to 10^{-9}, and the transient was repeated for $L_{iter} = 1, 2, \ldots, 15$. Total CPU time as a function of L_{iter} is shown in Figure 11, along with the total number of preconditioners required throughout the course of the transient. Though $L_{iter} = 6$ yields the minimum total CPU time for this particular transient, it appears that values ranging from 6 to 11 are reasonable. Note that all values of $L_{iter} \geq 12$ are equivalent, since only two preconditioners are obtained: one at the start of the transient and one just after the event at $t = 0.15$ seconds. The fact that the curve is not smooth is not surprising since that relationship between CGS, the preconditioner, and time-step control in SWIRL is quite complex.

5.2. SWIRL/CGS7$_b$ vs. SWIRL/BTD$_{inv}$. When the optimal choice of L_{iter} is used, the simulation is completed in only 235 seconds, or 1.8 times faster than SWIRL/BTD$_{inv}$ (even the worst choice for the iteration limit, $L_{iter} = 1$, is superior to SWIRL/BTD$_{inv}$, which requires over 421 seconds to model the transient). The CPU time used for each time step in the simulation
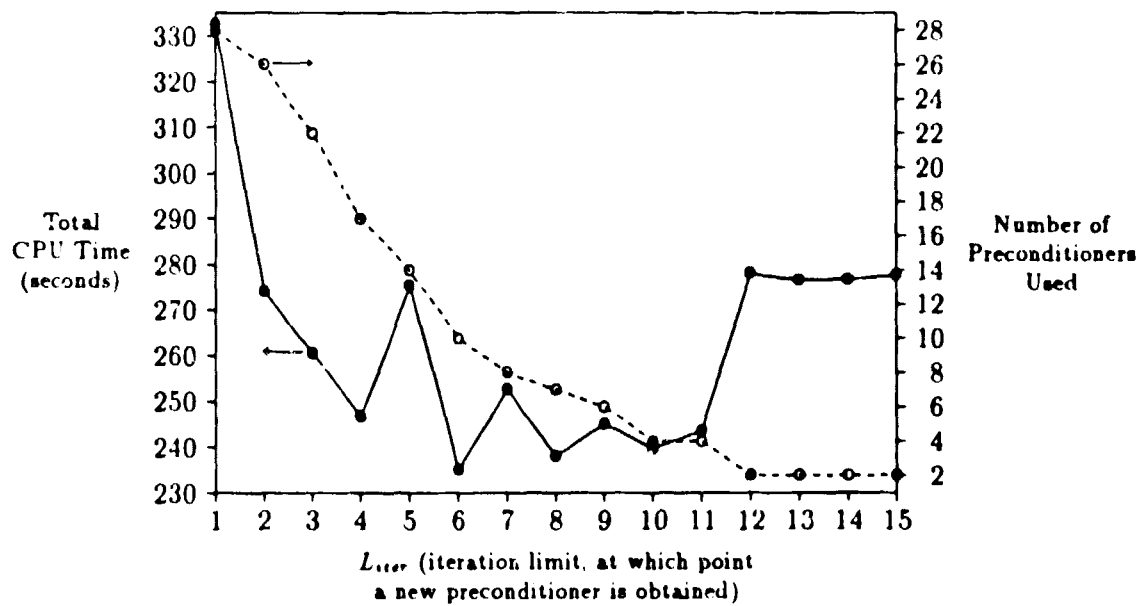
FIG. 11. *SWIRL/CGS7$_b$ total CPU time and number of preconditioners required during transient described in Section 4.2 for $\epsilon = 10^{-6}$ and various iteration limits.*

is shown in Figure 12 for SWIRL/BTD$_{inv}$ and for SWIRL/CGS7$_b$. Note that the CPU time for SWIRL/BTD$_{inv}$ is constant unless an attempted advancement fails and must be repeated with a smaller time step. The effect on SWIRL/CGS7$_b$ of obtaining a new preconditioner is clearly shown, particularly at step number 27, where the CPU time per time step drops significantly following the computation of a new preconditioner. Interestingly, SWIRL/CCS7$_b$ completes *every* time step faster than SWIRL/BTD$_{inv}$, even at the time of the event.

The superiority of SWIRL/CGS7$_b$ is even more pronounced on a less severe transient. Figure 13 shows the four CE vector CPU time per time step on the steady-state test problem. Note that for the 81-subchannel problem, SWIRL/CGS7$_b$ is approximately 3.5 times faster than SWIRL/BTD$_{inv}$.

6. Conclusions. This work demonstrates that CG-like methods can be efficient and robust for reactor thermal hydraulic simulations. Nevertheless, much work remains. In particular, the following areas are currently under investigation:

1. Other preconditioners, including standard incomplete LU-decomposition and variations on PBLU such as PBILU (in which only part of a past factorization is used) are being implemented and compared with PBLU.

2. Other CG like methods such as GMRES and ORTHOMIN are being implemented and compared with CGS.

3. A complete investigation of the eigenvalue spectrum of the coefficient matrices generated by this problem is underway. One concern is the effect of changing the implicitness of the temporal discretization or implementing a Courant violating scheme such as the stability enhancing two step method (SETS) [24] would have on the eigenspectrum. This information will aid in understanding the performance of various CG like methods.
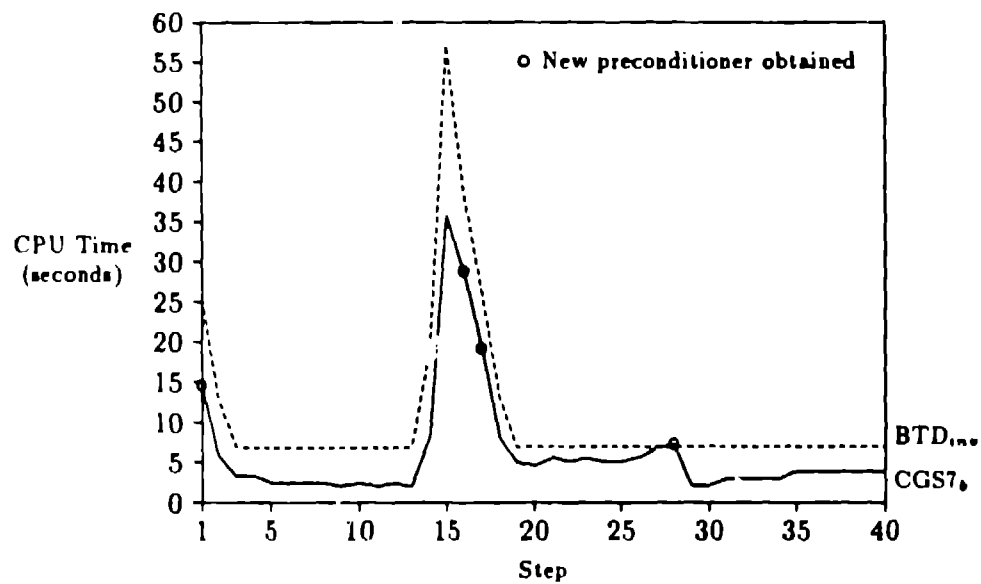
FIG. 12. $SWIRL/BTD_{inv}$ and $SWIRL/CGS7_b$ CPU time required for each step of transient described in Section 4.2, with $\epsilon = 10^{-9}$ and $L_{iter} = 11$.
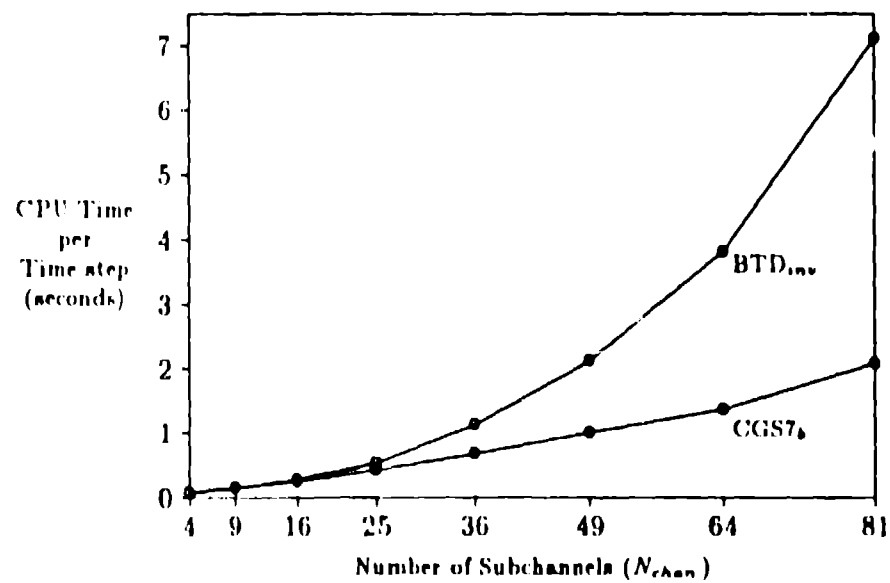

FIG. 13 $SWIRL/BTD_{inv}$ and $SWIRL/CGS7_b$ $(\epsilon = 10^{-9}, L_{iter} = 11)$ four CE vector CPU time per time step for the steady state problem described in Section 4.1 on an Alliant FX/4 for $N_{az} = 48$.

7. Nomenclature.

A_c	lateral cross sectional flow area
A_x	axial cross sectional flow area
A_{r_x}	axial flow area of gap–centered computational cell
D_H	hydraulic diameter
$f_{1\phi}$	single phase friction factor
g_z	gravitational constant
j_D^g	drift flux
k	axial forms loss coefficient
P	pressure
P_w	wetted perimeter
q_w	wall heat transfer rate
Δr	distance between adjacent subchannel centroids
Δt	time step
u	internal energy
V	axial velocity
v	velocity in arbitrary direction
w	lateral velocity
Δz	axial computational cell length
α	void fraction
ρ	density
σ	shear stress
τ_w	wall shear stress

Subscripts:

i	subchannel of interest
j	adjacent subchannel
ij	gap between subchannels i and j (in some instances also implies direction, i.e. , from subchannel i into subchannel j)
k	axial location
l	liquid phase (also occasionally used as a superscript)
g	vapor phase (also occasionally used as a superscript)
r	lateral quantity

Superscripts:

$t + \Delta t$	new time value
t^*	mixed implicit explicit quantity
$\ast$	laterally donored quantity

Other Notation:

$\sim$	axially donored quantity
$\sum_j$	summation over all adjacent subchannels

of the first author's graduate committee, for numerous suggestions and for inspiring the idea of reusing preconditioning matrices.

REFERENCES

[1] O. AXELSSON, *Conjugate gradient type methods for unsymmetric and inconsistent systems of linear equations*, Lin. Alg. Appl., 29 (1980), pp. 1-16.

[2] R. BOWRING AND P. MORENO, *COBRA-IIIC/MIT Computer Code Manual*, MIT for EPRI, Mar 1976.

[3] R. W. BOWRING, *HAMBO: A Computer Programme for the Subchannel Analysis of the Hydraulic and Burnout Characteristics of Rod Clusters*, United Kingdom Atomic Energy Authority, Winfrith, Dorchester, England. Part 1: General Description, AEEW-R524 (1967), Part 2: The Equations, AEEW-R582 (1968).

[4] H. CHELEMER, J. WEISMAN, AND L. S. TONG, *Subchannel thermal analysis of rod bundle cores*, Nucl. Eng. and Design, 21 (1972), p. 3.

[5] C.-J. B. CHEN, T. H. CHEN, AND W. T. SHA, *BODYFIT-2PE-HEM: LWR Core Thermal-Hydraulic Code Using Boundary-fitted Coordinates and Two-Phase Homogeneous Equilibrium Model*, Argonne National Laboratory, Aug 1985. EPRI NP-3768-CCM, Project 1383-1.

[6] C.-J. B. CHEN, W. T. SHA, M. L. DORIA, R. C. SCHMITT, AND J. F. THOMPSON, *BODYFIT-1FE: A Computer Code for Three-Dimensional Steady-State / Transient Single-Phase Rod-Bundle Thermal-Hydraulic Analysis*, Argonne National Laboratory, Nov 1980. NUREG/CR-1874, ANL-80-127.

[7] G. R. DI BROZOLO AND Y. ROBERT, *Parallel conjugate gradient-like algorithms for solving sparse nonsymmetric linear systems on a vector multiprocessor*, Parallel Computing, 11 (1989), pp. 223-239.

[8] H. M. DOMANUS, R. C. SCHMITT, W. T. SHA, AND V. L. SHAH, *COMMIX-1A: A Three-Dimensional Transient Single-Phase Computer Program for Thermal-Hydraulic Analysis of Single and Multicomponent Systems*, Argonne National Laboratory, Dec 1983. NUREG/CR-2896, ANL-82-25.

[9] H. M. DOMANUS, W. T. SHA, AND V. L. SHAH et al, *COMMIX-2: A Steady/Unsteady Single-Phase/Two-Phase Three-Dimensional Computer Program for Thermal-Hydraulic Analysis of Reactor Components*, Argonne National Laboratory, Oct 1980. NUREG/CR-1807, ANL-81-10.

[10] J. M. DOSTER, *Numerical solutions of multiphase flow problems*, in Simulators VI, The Society for Computer Simulation International, 1989, pp. 59-66.

[11] DUKE POWER COMPANY, *McGuire Nuclear Station Final Safety Analysis Report*.

[12] EG&G IDAHO, INC., *RELAP5/MOD1 Code Manual, Volume 1: System Models and Numerical Methods*, 1980.

[13] M. FAJEAU, *Programme FLICA: Etude Thermodynamique d'un Reacteur ou d'une Boucle d'Essai*, Centre d'Etudes Nucleaires, Saclay, France, 1969. CEA-R-3716.

[14] R. FLETCHER, *Conjugate Gradient Methods for Indefinite Systems*, Lecture Notes in Mathematics 506, Springer-Verlag, New York, 1976.

[15] T. L. GEORGE et al., *COBRA-WC: A Version of COBRA for Single-Phase Multi-Assembly Thermal-Hydraulic Transient Analysis*, Pacific Northwest Laboratory, Richland, WA, July 1980. PNL-3259.

[16] D. HOWARD, W. M. CONNOLLEY, AND J. S. ROLLETT, *Unsymmetric conjugate gradient methods and sparse direct methods in finite element flow simulation*, Int. J. Num. Meth. Fluids, 10 (1990), pp. 925-945.

[17] K. C. JEA AND D. M. YOUNG, *Generalized conjugate-gradient acceleration of nonsymmetrizable iterative methods*, Lin. Alg. Appl., 34 (1980), pp. 159-194.

[18] ———, *On the simplification of generalized conjugate-gradient methods for nonsymmetrizable linear systems*, Lin. Alg. Appl., 52/53 (1983), pp. 399-417.

[19] P. JOLY AND R. EYMARD, *Preconditioned biconjugate gradient methods for numerical reservoir simulation*, J. Comp. Phys., 91 (1990), pp. 298-309.

[20] H. P. LANGTANGEN AND A. TVEITO, *A numerical comparison of conjugate gradient-like methods*, Comm. Appl. Numer. Meth., 4 (1988), pp. 793-798.

[21] D. R. LILES AND W. H. REED, *A semi-implicit method for two-phase fluid dynamics*, J. Comp. Physics, 26 (1978), pp. 390-407.

[22] J. LOOMIS, W. H. REED, A. SCHOR, H. B. STEWART, AND L. WOLF, *THERMIT: A Computer Program for Three-Dimensional Thermal-Hydraulic Analysis of Light-Water-Reactor Cores*, Massachusetts Institute of Technology, Cambridge, MA, Sept 1981. EPRI NP-2023, Project 516.

[23] LOS ALAMOS NATIONAL LABORATORY, *TRAC-P1A: An advanced best estimate computer program for PWR LOCA analysis*, May 1979.

[24] J. H. MAHAFFY, *A stability-enhancing two-step method for fluid flow calculations*, J. Comp. Phys., 46 (1982), p. 329.

[25] R. POTTER et al, *SABRE 1 A Computer Program for the Calculation of Three-Dimensional Flows in Rod Clusters*, July 1976. AEEW-R 1057.

[26] D. S. ROWE, *Crossflow Mixing Between Parallel Flow Channels During Boiling, Part 1: COBRA - Computer Program for Coolant Boiling in Rod Arrays*, Pacific Northwest Laboratory, Richland, WA, Mar 1967. BNWL-371, pt 1.

[27] ———, *COBRA-II: A Digital Computer Program for Thermal Hydraulic Subchannel Analysis of Rod Bundle Nuclear Fuel Elements*, Pacific Northwest Laboratory, Richland, WA, Feb 1970. BNWL-1229.

[28] ———, *COBRA-III: A Digital Computer Program for Steady-State and Transient Thermal-Hydraulic Analysis of Rod Bundle Nuclear Fuel Elements*, Pacific Northwest Laboratory, Richland, WA, 1971. BNWL-B-82.

[29] ———, *COBRA-IIIC: A Digital Computer Program for Steady-State and Transient Thermal-Hydraulic Analysis of Rod Bundle Nuclear Fuel Elements*, Pacific Northwest Laboratory, Richland, WA, Mar 1973. BNWL-1695.

[30] Y. SAAD, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comp., 37 (1981), pp. 105-126.

[31] ———, *Krylov subspace methods on supercomputers*, SIAM J. Sci. Stat. Comput., 10 (1989), pp. 1200-1232.

[32] Y. SAAD AND M. H. SCHULTZ, *GMRES: A generalised minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856-869.

[33] W. T. SHA, *An Overview of Rod-Bundle Thermal-Hydraulic Analysis*, Argonne National Laboratory, Nov 1980. NUREG/CR-1825, ANL-79-10.

[34] W. T. SHA, H. M. DOMANUS, R. C. SCHMITT, J. J. OHAS, AND E. I. H. LIN, *COMMIX-1: A Three-Dimensional, Transient, Single-Phase Component Computer Program for Thermal-Hydraulic Analysis*, Argonne National Laboratory, Jan 1978. NUREG/0415, ANL-77-96.

[35] ———, *A new approach for rod-bundle thermal-hydraulic analysis*, Nucl. Tech., 46 (1979), pp. 268-280.

[36] W. T. SHA AND J. F. THOMPSON, *Rod Bundle Thermal-Hydraulic Analysis using Boundary-Fitted Coordinate System*, Argonne National Laboratory, Jan 1979. NUREG/CR-0001, ANL-78-1.

[37] P. SONNEVELD, *CGS, A fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 10 (1989), pp. 36-52.

[38] P. SONNEVELD, P. WESSELING, AND P. M. ZEEUW, *Multigrid and conjugate gradient methods as convergence acceleration techniques*, in Multigrid Methods for Integral and Differential Equations, D. J. Paddon and H. Holstein, eds., Oxford University Press, Oxford, 1985, pp. 117-167.

[39] C. W. STEWART, J. M. CUTA, A. S. KOONTZ, J. M. KELLY, K. L. BASEHORE, T. L. GEORGE, AND D. S. ROWE, *VIPRE Code Manual: Volume 1, Models and Constitutive Relations*, Battelle Pacific Northwest Laboratories, for Electric Power Research Institute, Richland, WA, June 1983. Research Project 1584.

[40] C. W. STEWART, J. M. CUTA, S. D. MONTGOMERY, J. M. KELLY, K. L. BASEHORE, T. L. GEORGE, AND D. S. ROWE, *VIPRE-01: A Thermal-Hydraulic Code for Reactor Cores; Volume 1: Mathematical Modeling (Revision 2)*, Battelle Pacific Northwest Laboratories, for Electric Power Research Institute, Richland, WA, July 1985. EPRI NP-2511-CCM, Volume 1, Revision 2, Research Project 1584-1.

[41] J. A. TURNER, *Transient, Two-Phase Subchannel Analysis on Advanced Architecture Computers*, PhD thesis, North Carolina State University, Raleigh, N. C., 1990. Department of Nuclear Engineering.

[42] C. L. WHEELER et al., *COBRA-IV-I: An Interim Version of COBRA for Thermal-Hydraulic Analysis of Rod Bundle Nuclear Fuel Elements and Cores*, Pacific Northwest Laboratory, Richland, WA, Mar 1973. BNWL-1962.

[43] W. ZERNICK, H. B. CURRIN, E. ELYASH, AND G. PREVITI, *THINC - A Thermal Hydrodynamic Interaction Code for a Semi-Open or Closed Channel Core*, Feb 1962. WCAP-3704.

[44] N. ZUBER AND J. A. FINDLAY, *Average volumetric concentration in two-phase flow systems*, J. Heat Trans., (1965), pp. 453-468.